MEDINFO 2015: eHealth-enabled Health
I.N. Sarkar et al. (Eds.)

doi:10.3233/978-1-61499-564-7-227

Product-based Safety Certification for Medical Devices Embedded Software

José Augusto Neto[a], Jemerson Figueiredo Damásio[a], Paul Monthaler[a], Misael Morais[a]

[a] *State University of Paraíba – UEPB, Campina Grande, Paraíba, Brazil*

Abstract

Worldwide medical device embedded software certification practices are currently focused on manufacturing best practices. In Brazil, the national regulatory agency does not hold a local certification process for software-intensive medical devices and admits international certification (e.g. FDA and CE) from local and international industry to operate in the Brazilian health care market. We present here a product-based certification process as a candidate process to support the Brazilian regulatory agency ANVISA in medical device software regulation. Center of Strategic Technology for Healthcare (NUTES) medical device embedded software certification is based on a solid safety quality model and has been tested with reasonable success against the Class I risk device Generic Infusion Pump (GIP).

Keywords:

Medical Devices; Embedded Software Certification; Safety Certification; Certification Process.

Introduction

The software embedded in medical devices introduces new features and defines competitive differences among same-class equipment. Despite this benefit, the presence of software, and its intense interaction with electrical and electronic components, implies new malfunctioning risks and potential harm to patients and healthcare professionals who interact with it. In the last decade, the US regulatory agency (FDA) has published information concerning fault cases on medical devices and the reports show 1,154,451 cases of damage, as well as the need for 5,294 recalls [1]. The same study points out that software is still largely responsible for recalls in medical devices, and is directly involved in 33.3% of recalls for Class I (high risk) equipment, 65.6% of Class II (medium risk) and 75.3% of Class III (low risk).

The FDA provides some guidance to the industry on the general principles of software validation. In a specific document [2], the FDA specifies that software must meet a set of rules on the production process, as defined in the 21 CFR §820.70 law [3]. The legislation in Brazil is less specific on medical device embedded software, but recently the Brazilian regulatory agency ANVISA exposed a new norm to public consultation [4] which, in short, proposes to comply with the FDA production-based approach.

Overall, the current certification models are mainly based on ensuring that the production processes follow international standards such as ISO 14971 [5] and ISO 80002 [6]. Promising alternatives are known, such as the argument-based approach, where manufacturers provide safety cases and certification authorities assess them individually, and the safety quality model approach, in which quality questions are provided to guide assessors to decide whether a software product is safe or not [7]. This work extends the NUTES-IESE/Fraunhofer Software Safety Quality Model [7] with a safety certification process based on it.

The Product Certifier Body (OCP in Portuguese) is the group appointed by ANVISA that have expertise to audit medical device production processes. However, to the best of our knowledge, there is no OCP accredited to perform specific certification for embedded software in medical devices, even if the focus is on the production process.

NUTES Quality Model and the Certification Process Model

The Center for Strategic Technologies in Healthcare (NUTES) was created in 2011, as result of a partnership between the State University of Paraíba (UEPB – www.uepb.edu.br) and the Brazilian Health Ministry, aiming to develop products and technologies to support the Brazilian medical device industry, as well as the regulatory agencies that supervise it.

In 2013 and 2014, the NUTES technical team, in cooperation with the German institute IESE/Fraunhofer-Kaiserlautern, developed a safety quality model for medical device embedded software totally centered on the final product, instead of the manufacturing best practices. The basis of the NUTES Quality Model is a strong theoretical framework of safety engineering, combined with the Fraunhofer background, which has been accumulated through many years of providing services to industries in which embedded software is critical, such as automotive and the airline industry [8].

The NUTES Quality Model proposes asking a set of quality questions against the software documentation; those questions precisely define what is considered safe software in a medical device [7]. In spite of that, the model does not include a process for how these questions should be asked, nor does it provide a step by step tutorial to check whether a software is safe or not.

This paper presents a structured process used at NUTES to certify embedded software in medical devices. The process is original to Brazil and is not production-based, but rather product-based. It uses the NUTES Quality Model and, as a consequence, it is reusable and adaptable to any software-intensive medical device. This is possible due to the quality model structure, based on general safety cases, which in turn refers to requirements, architecture, testing, and code aspects (common to every software product).

The certification process presented in this work can be, for didactic purposes, split into three phases: 1) the initial customer interaction; 2) the documents acquisition to support the certification realization; and 3) the software analysis. Particularly for phases 2 and 3, we list the input and output artifacts. We also present the criterion of division in 5 subareas of technical knowledge (safety, code, tests, architecture and usability), and the aggregation rules between areas, which produces a quantitative and qualitative outcome of the certification.

The provided process is compatible with the technical requirements demanded by ANVISA, and is continuously evolving to stay aligned with technical requirements

Pilot Evaluation

Manufacturers resist collaborating in experimental assessments since NUTES certification exposes classified product features. Moreover, ANVISA still does not require a local software certification. As an alternative, we performed a pilot evaluation of the process applying it to certify the Generic Infusion Pump (GIP) [9]. The GIP is a project developed by the FDA Software Engineering Laboratory in cooperation with the University of Pennsylvania.

The GIP failed the NUTES certification process, mostly because we could not satisfy part of the quality model, either by lack of documents or due to real product safety issues. At the end of this document, we present a proper discussion on the results, implications and limitations of this approach.

Methods

Theoretical Background

The process described here results from a bibliographic review of documents provided by regulatory agencies that describe the main product certification steps [10, 11]. Moreover, the NUTES certification team is working on the ISO 17025 [12] certification pipeline; and in planning the next step, the ISO 9001 [13].

The process described next is documented within a full quality management system, compliant with international standards. We built the process assisted by an ISO consulting company. The interaction between NUTES and this company was formal, and occurred during eight months in 2014, from March to November. Furthermore, there are several direct and indirect contributions of members of the FDA and ANVISA to the current version of the certification process. The latter is much closer and especially important since many ANVISA members also act as professors at NUTES Postgraduate Program in Science and Technology in Health. The result of this effort is a structured model, presented in Business Process Modeling Notation (BPMN), and a list of documents that support its execution.

We executed the GIP certification process in September 2014, in a 15-day period, and it was attended by the NUTES Software and Safety Engineering Research Team. Currently, this team is composed of PhD Computer Science faculty members with relevant contributions in the software engineering field [7, 14, 15, 16]. The team has been through a two-year safety-focused training by means of a cooperation with the experienced IESE/Fraunhofer Kaiserlautern.

Evaluation Method

We primarily evaluated the process in a pilot experiment for the GIP certification. We consider the infusion pump choice adequate for the following specific reasons:

1. The FDA claims [1] that infusion pumps are a continuous source of safety problems;
2. The GIP in an open specification. This is interesting for future evaluation and replication of the efforts in next phases of this work. Some obstacles could emerge if we had opted for a proprietary device.
3. The GIP is a robust project of a regulatory agency and its documentation works as benchmark for both manufacturers and the academic community.
4. It is a popular project, cited in more than 10 academic papers.
5. Its conception is safety-focused, therefore directly addresses the requirements of this work.

We used the GIP publicly available documentation [17] as input to the certification process; it includes articles, models and code. Complementary documentation was required and produced by means of NUTES internal efforts [18]. As a result, we present an argument-based validation and discuss the impact of the input documentation on the process.

Results

Certification Process Model

The certification process entails three distinct and sequenced phases: **Phase 1 (Business agreement)** – contains the customer-NUTES interactions up to hiring the certification. In this phase, negotiation and contract signature occur; **Phase 2 (Document submission)** – comprises the actions performed with the purpose to acquire the artifacts required to execute the certification process. This phase is interactive and incremental (i.e., we held document evaluation rounds and additional documentation is requested when necessary). The last phase takes place when the interactions end and the documentation is complete; **Phase 3 (Software analysis)** – the main certification phase. We share the documentation among the teams in charge of subareas of certification. The subarea certification is performed, and the individual results are composed into a final certification result.

The brief description that follows summarizes the knowledge recorded in 19 operational processes and 15 document templates. The big picture of this work includes the link between this certification process model and a quality management system, together they configure the minimum elements required by ISO 17025 and ISO 9001, for which NUTES is to be submitted.

Figure 1 details the certification process using BPMN. Notice that Phase 1 is not present in the flow since it is all about commercial interaction between NUTES and the product manufacturer, or ANVISA (both are potential clients for embedded software certification). This interaction includes defining the risk class of the equipment and the embedded software under certification, as well as the milestones and deadlines, specially related to the current specification of the NUTES technical team.

Composing the Certification Dossier

The certification process starts in Phase 2, in which the software documentation is requested from the client. This request specifies deadlines and delivery orders for the documentation. We analyze all documents received and depending on its completeness for the type of device under evaluation, we decide whether the received documentation meets the process needs; if it does not, we send a new request. The input software documents for certification are:

1. Test Plans – a document that guided the execution of different categories of tests in the system;
2. Test Logs – a detailed record that includes time marks of the execution tests results;
3. Test Reports – a formal document that describes the tests results, where successes and failures are presented;
4. Architectural Document – a high-level view of the system. It uses text and architectural models to

explain the system in a macro vision (e.g.: which module interacts with the alarm module);

5. Functional Specification – presents the system expected behavior;
6. Safety Requirements Document – this document lists the body of functionalities directly related to guarantee the safety of the system;
7. Software Design – it describes the system in a still abstract perspective, but in a much lower level than the architecture (e.g.: how small code unities the interaction that occurs inside the alarm module);
8. Software Source Code and Support Libraries – the system itself, coded in a known programming language;
9. Usability Engineering Document – it describes the decision behind the way the interaction between system and user is considered;
10. User Manual – a document addressed to the final user. It explains the correct way one should act in order to interact with the system;
11. Alarms Specification – it describes the events that might happen, and how they fire observable notifications to the user;
12. Risk Management File – an important document, suggested by ISO 80002 [6], which exposes every safety aspect of a device and how they are considered (e.g.: known risks, fail occurrence chance for different categories, etc.).

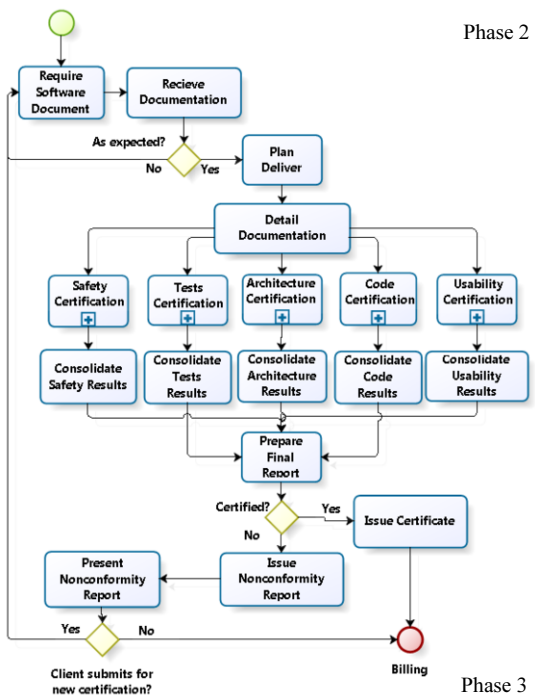

Figure 1 – The certification process for medical devices embedded software

Notice that these documents are a subset of those required by the FDA. This indicates that their production is typical, and does not impose drastic changes to the way that the software industry currently works. Moreover, in the case of certifying a real software instead of an open specification, the manufacturer is provided with "The NUTES Guide for Certification of Embedded Software for Clients". The guide goes into detail on each required document and also presents sample documents with the completeness level expected for the certification process.

Phase 2 takes at least one day and at most a week; this variation depends on the customer documentation availability. Once documentation is accepted, the third phase starts.

In Phase 3 we establish a delivery plan for each team (i.e., safety, test, architecture, code and usability) involved in the certification. This plan takes into account the certifications in progress and the teams' availability. Next, the full documentation body is detailed and shared (with possible copies) for the subareas. Finally, they generate specific reports.

Figure 2 presents the process executed in each subarea. The first task is about receiving documents and checking if the documentation is complete. At this point, any team can make requests for additional documentation to the certification manager who controls the documents originally received in Phase 2.

The subarea planning certification represents the first interaction with the NUTES Quality Model. The first step is about defining the set of questions that apply to the device under certification. At this point, teams should select questions and classify each of them as mandatory, desirable or optional according to device's risk class. In the second step, we define deadlines for the execution and report delivery. All steps, including the choice of questions and their classification, demand prior technical background and should be endorsed by a previous certification executed at NUTES.

The execution task refers to the assessment of available documentation seeking answers to the selected questions. For certification purposes, the NUTES Quality Model's questions must receive positive answers. Negative responses indicate a possible non-compliance of the software.

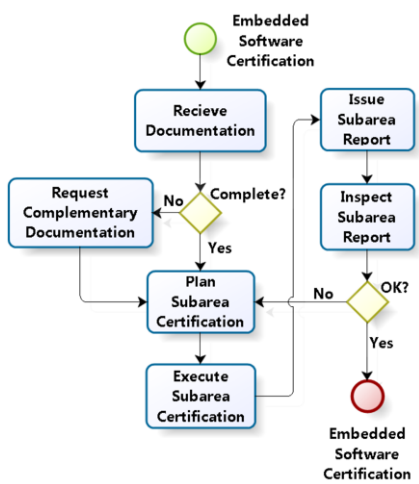

Figure 2 – The subarea certification process for medical devices embedded software.

In the end, each area produces two reports: a qualitative description of the product, and a quantitative result of the subarea certification. The first shows the assessed documentation, and presents a textual argument about the strengths and weaknesses of the product, highlighting nonconformities that led to negative answers (if any). Finally,

if the software fails the certification, the report suggests modifications to the software, such as new features to enable a better score on future certifications. The quantitative report describes the number of positive answers for each class of questions in comparison to the maximum score per class. Figure 3 shows an example of part of a quantitative subarea report. We express the number of positive and negative responses for each class in both textual and graphical representations.

Mandatory		Desirable		Optional	
Yes	No	Yes	No	Yes	No
15	0	3	2	1	1
100%	0%	60%	40%	50%	50%

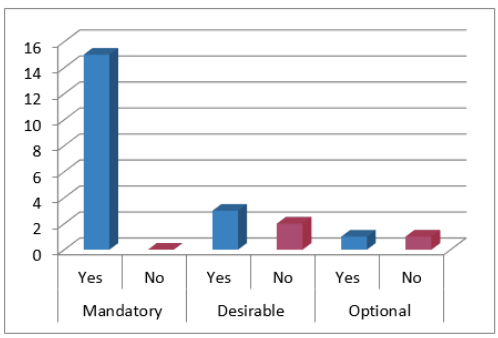

Figure 3 – Subarea quantitative results showing the total number of positive and negative answers for each class.

Finally, the team leader inspects the subarea report looking for mistakes or inconsistencies. If the team members finds any divergence, they must revisit the whole subarea process.

Returning to the main process, the certification manager analyzes each subarea report and produces the final report. The final report consists, first, of all subareas qualitative observations presented in a coherent outline. Next, it includes the subareas quantitative reports in a summarized table, with both "Yes" and "No" responses for each class of questions. To achieve the final result, the following aggregation rule is applied: questions related to mandatory safety features have a limit of zero negative answers allowed. Questions related to optional and desired safety features have an upper limit for negative answers according to the device's risk class. For the GIP, two is the limit for negative answers in desirable questions, and a varying number of negative answers is acceptable for optional safety features related questions. Each area specialist, based on international standards practices and individual expertise, defines those bounds.

Once the report is ready, one of the following situations occurs: 1) A certification is assembled and sent to the customer if it is compliant (i.e.: negative answers did not go over the limits) and therefore the embedded is considered safe; or 2) The customer is told the software failed the certification and is provided with the report that clarifies the problems that led to the negative result.

Pilot Evaluation

We applied the certification process to the GIP and the results show that the device, as currently described in the public specification, is not safe as a software intensive medical device is required to be according to NUTES Quality Model and its 300 safety-related quality questions.

However, due to confidentiality reasons, we cannot elaborate on the real results for any subarea but Safety. Such restriction does not influence the certification process comprehension, the main goal of this work. Table 1 presents the quantitative results, where only Safety results are accurate while others are slightly modified. We altered these results in a way that conclusions are the same from the original report, as follows.

Table 1 – Quantitative result of GIP certification against NUTES Quality Model questions

Subarea	Mandatory		Desirable		Optional	
	Yes	No	Yes	No	Yes	No
Usability	3	10	12	16	14	1
Safety	20	5	1	0	0	0
Architecture	5	10	17	1	2	10
Requirements	18	3	36	5	6	20
Code	15	0	4	1	1	1
Test	4	36	11	6	3	3
Total	65	64	81	29	26	35

A full "Yes" score is a must for Mandatory questions on every subarea; for the Safety subarea (real result), GIP's 20 "Yes" out of 25 questions is enough to fail the certification. The illustrative numbers used for the remaining subareas show that we address Usability, Architecture, Tests and Requirement's safety issues properly, and Tests holds the weakest safety treatment. On the other hand, Code alone had the best result, mainly because GIP's code was a good source, where we found positive answers for all questions.

Discussion

The GIP project is largely different from conventional software. Its documentation derives from multiple academic works, while conventional documentation is generally produced in sequence, and each document is clearly related to others during the software engineering process. In this paper, we have claimed that we used all the available documentation for GIP. The first problem is that the documentation found was not enough to fill out the required list. For example, we used the Hazard Analysis [19] and a document of Risk Analysis [18] to address the lack of a Risk Management File. Therefore, from the 64 negations for mandatory question, only 14 were based on real evidence. The other 50 were explained in terms of "No evidence in the documentation".

In a real product scenario, the information scarcity is also possible, but we would be able to ask the customer for extra documentation. Another aspect is the documents internally produced by NUTES team to reach the documentation required by the process, we believe that the certification scores could be higher if extra effort was placed in better composing of mock documents. Additional in-depth considerations, specifically for the Architecture subarea, can be found in another work [14].

Conclusion

The regulation for embedded software in medical devices in Brazil is currently based on international certifications, such as the FDA or CE. These certifications focus on the audit of manufacturing processes, assuming that well-executed processes will generate safe products, but this approach has problems. In this work we presented a certification process focused on the product, supported by a quality model that provides a general safety case for embedded software in medical devices.

We applied the designed process to the GIP and the results point to the high standards to which a software-intensive medical device must comply with to be considered safe, and therefore certified. We have fully executed the process, but real product certifications are necessary to better validate and improve the process proposed here. First agreements with industry partners have already taken place, and we intend to execute new rounds of the process on commercialized products soon.

The certification process is currently undergoing ISO 17025 certification, and shall be submitted to the ISO 9001 soon. A first internal audit phase for ISO 17025 has already been accomplished. This step fulfills accreditation requirements to NUTES be able to provide ANVISA with a local software certification service to Brazilian and global manufacturers.

Acknowledgements

This research is directly supported by NUTES/UEPB and several of its researchers, who have made multiple contributions in the pilot evaluation results. We also thank Dr. Kleber Nobrega and his team at Perceptum, who offered strong methodological support for us to create and refine the certification process presented here.

References

[1] Alemzadeh, H.; Iyer, R.K.; Kalbarczyk, Z.; Raman, J. Analysis of Safety-Critical Computer Failures in Medical Devices. IEEE Computer Society 2013;11:14–26.

[2] FDA, General Principles of Software Validation; Final Guidance for Industry and FDA Staff Health. San Francisco 2002; 47.

[3] FDA, 21CFR820.70. [online: http://www.accessdata.fda.gov/scripts/cdrh/cfdocs/cfcfr/CFRSearch.cfm?fr=820.70. Accessed 2014 Nov 3rd].

[4] Portaria 407, Consulta pública sobre o Aperfeiçoamento dos Requisitos de Avaliação da Conformidade para Equipamentos sob, 2014 August 26th. [online: http://www.inmetro.gov.br/legislacao/rtac/pdf/RTAC002160.pdf. accessed: 2014 Nov 16th]

[5] ISO/TC 210, Medical devices - Application of risk management to medical devices, 2010, ISO/TR 14971:2007.

[6] ISO/TC 210, Medical devices software – Part 1: Guidance on the Application of ISO 14971 to medical device software, 2009, ISO/TR 80002:2009.

[7] Adler, R.; Kemmann, S.; Filho, D. M.; Augusto, J. Safety assessment of software-intensive medical devices. Proceeding of IEEE Symposium on Software Reliability Engineering Workshops 2013;217-22.

[8] Fraunhofer Institute Research Topics - Transportation and Mobility [online: http://www.fraunhofer.de/en/research-topics/transportation-mobility.html. Access: 2014 Dec 3rd].

[9] The generic patient controlled analgesia pump model. [online: http://rtg.cis.upenn.edu/gip.php3. accessed: 2014 Nov 15th]

[10] FDA Overview of Device Regulation, 2014. [online: http://www.fda.gov/MedicalDevices/DeviceRegulationandGuidance/Overview/. Access: 2014 Nov 12th]

[11] Europe CE Approval Process for Medical Devices, 2014 [online:http://www.emergogroup.com/pt/resources/europe-process-chart. Access: 2014 Nov 11th]

[12] ISO/CASCO, General requirements for the competence of testing and calibration laboratories, 2010, ISO/TR 17025:2005.

[13] ISO/TC 176/SC 2, Quality Management Systems - Requirements, 2008, ISO/TR 9001:2008.

[14] Leite, F.; Antonino, P.; Barbosa, P.; Kemmann, S.; Mendonca, R. Are the Current Architectural Practices Suitable for Safety Aspects of Medical Devices? An Exploratory Investigation. IEEE HealthCom, 2014.

[15] Costa, Túlio H. ; Andrade, Rodrigo ; Maciel, Edna Queiroz; Lima, Lukas Teles ; Bublitz, Frederico Moreira . A Safety Engineering Hazard Identification for Hemodialysis Systems. In: 2015 International Conference on Consumer Electronics, 2015, Las Vegas. International Conference on Consumer Electronics, 2015. p. 86-87.

[16] Silva, L. C.; Perkusich, M.; Bublitz, F. M. ; Almeida, H.; Perkusich, A. A model-based architecture for testing medical cyber-physical systems. In: the 29th Annual ACM Symposium, 2014, Gyeongju. Proceedings of the 29th Annual ACM Symposium on Applied Computing - SAC '14. p. 25-30.

[17] The Generic Infusion Pump publications [online: http://rtg.cis.upenn.edu/gip_pub.php3. Acessed: 2014 Nov 14th]

[18] NUTES GIP Documentation [online: https://www.dropbox.com/sh/pbz4qncfokvc938/AABPRIOckiluQfzcYHH1Doepa?dl=0. Access: 2014 Dec 7th]

[19] Yi Z.; Paul J.; Raoul J.A Hazard Analysis for a Generic Insulin Infusion Pump. Journal of Diabetes Science and Technology, March 2010, 4 (2):263-83.

Address for correspondence

José Augusto de Oliveira Neto, E-mail: zedeguga@gmail.com, Baraúnas st. 351 - Bairro Universitário - Campina Grande/PB - Brazil, Zip 58429-500